Exhibit D
'tzlbac14.log'

```
SQL> @tzlbac14
SQL>
SQL> CONNECT SCOTT/TIGER
Connected.
SQL>
SQL>    CREATE TABLE abc
  2      (COL1 VARCHAR2(45));

Table created.

SQL>
SQL>    CREATE TABLE jing
  2      (COL1 VARCHAR2(45));

Table created.

SQL>
SQL>    GRANT ALL ON abc TO LBACSYS;

Grant succeeded.

SQL>
SQL> -- This should not be allowed as the user is SCOTT
SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('simple','lbac$test1','raghu');
BEGIN LBAC_SYSDBA.CREATE_POLICY('simple','lbac$test1','raghu'); END;

      *
ERROR at line 1:
ORA-06550: line 1, column 7:
PLS-00201: identifier 'LBACSYS.LBAC_SYSDBA' must be declared
ORA-06550: line 1, column 7:
PL/SQL: Statement ignored


SQL>
SQL> CONNECT LBACSYS/LBACSYS
Connected.
[START: CLAIMS 1 and 21
Shows a plurality of label-based policies that are created in the database; 3$^{rd}$
parameter is the policy column name]
SQL>
SQL> -- Create policies in database
SQL>
SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('simple','lbac$test1','raghu');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('complex','lbac$test1','rghu');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('sile','lbac$test1');

PL/SQL procedure successfully completed.

[END: CLAIMS 1 and 21]
```

```
SQL>
SQL> -- Error Conditions
SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('complex','lbac$test1','aghu');
BEGIN LBAC_SYSDBA.CREATE_POLICY('complex','lbac$test1','aghu'); END;

*
ERROR at line 1:
ORA-12447: policy role already exists for policy complex
ORA-06512: at "LBACSYS.LBAC_STANDARD", line 0
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-01921: role name 'COMPLEX_DBA' conflicts with another user or role name
ORA-06512: at line 1


SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('dummy','lbac$test1');
BEGIN LBAC_SYSDBA.CREATE_POLICY('dummy','lbac$test1'); END;

*
ERROR at line 1:
ORA-12442: policy column "TESTLABEL" already used by an existing policy
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-06512: at line .*


SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('new','lbac$test1','raghu');
BEGIN LBAC_SYSDBA.CREATE_POLICY('new','lbac$test1','raghu'); END;

*
ERROR at line 1:
ORA-12442: policy column "RAGHU" already used by an existing policy
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-06512: at line .*


SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('new123','lbac$tt1','xyz');
BEGIN LBAC_SYSDBA.CREATE_POLICY('new123','lbac$tt1','xyz'); END;

*
ERROR at line 1:
ORA-12412: policy package lbac$tt1 is not installed
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-06512: at line .*


SQL>
SQL> -- This should not fail ...
SQL>    EXECUTE
LBAC_SYSDBA.CREATE_POLICY('abcdefghijklmnopqrstuvwxyz1234','lbac$test1','fdf');

PL/SQL procedure successfully completed.

SQL>
SQL>    EXECUTE
LBAC_SYSDBA.CREATE_POLICY('abcdefghijklmnopqrstuvwxyz','lbac$test1','fdf');
BEGIN
LBAC_SYSDBA.CREATE_POLICY('abcdefghijklmnopqrstuvwxyz','lbac$test1','fdf'); END;
```

```
*
ERROR at line 1:
ORA-12447: policy role already exists for policy abcdefghijklmnopqrstuvwxyz
ORA-06512: at "LBACSYS.LBAC_STANDARD", line 0
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-01921: role name 'ABCDEFGHIJKLMNOPQRSTUVWXYZ_DBA' conflicts with another
user or role name
ORA-06512: at line 1


SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('india','lbac$test1','fdfefg');

PL/SQL procedure successfully completed.

SQL>
SQL> -- Add 5 policies due to max_label_policies default increase from 5 to 10.
SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('ab1','lbac$test1','vc1');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('ab2','lbac$test1','vc2');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('ab3','lbac$test1','vc3');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('ab4','lbac$test1','vc4');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('ab5','lbac$test1','vc5');

PL/SQL procedure successfully completed.

SQL>
SQL> -- Error Conditions
SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('abc','lbac$test1','vcx');
BEGIN LBAC_SYSDBA.CREATE_POLICY('abc','lbac$test1','vcx'); END;

*
ERROR at line 1:
ORA-12422: max policies exceeded
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-06512: at line 1


SQL>    EXECUTE LBAC_SYSDBA.CREATE_POLICY('simple');
BEGIN LBAC_SYSDBA.CREATE_POLICY('simple'); END;

      *
ERROR at line 1:
ORA-06550: line 1, column 7:
PLS-00306: wrong number or types of arguments in call to 'CREATE_POLICY'
ORA-06550: line 1, column 7:
PL/SQL: Statement ignored
```

```
SQL>    EXECUTE
LBAC_SYSDBA.CREATE_POLICY('abcdefghijklmnopqrstuvwxyz12345','lbac$test1','ragu')
;
BEGIN
LBAC_SYSDBA.CREATE_POLICY('abcdefghijklmnopqrstuvwxyz12345','lbac$test1','ragu')
; END;

*
ERROR at line 1:
ORA-12447: policy role already exists for policy
abcdefghijklmnopqrstuvwxyz12345
ORA-06512: at "LBACSYS.LBAC_STANDARD", line 0
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-01921: role name 'ABCDEFGHIJKLMNOPQRSTUVWXYZ_DBA' conflicts with another
user or role name
ORA-06512: at line 1


SQL>
SQL> -- Drop extra 5 policies from above.
SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('ab1');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('ab2');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('ab3');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('ab4');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('ab5');

PL/SQL procedure successfully completed.
```

[START: CLAIMS 1 and 21 – list of policies created]

```
SQL> -- Let us check the policies created ...
SQL>    SELECT *
  2     FROM DBA_LBAC_POLICIES
  3     ORDER BY POLICY_NAME;
```

POLICY_NAME	COLUMN_NAME
---------------------------------	---------------------------------

PACKAGE	BIN_SIZE STATUS
---------------------------------	---------- --------

DEFAULT_FORMAT	POLICY_FORMAT
---------------------------------	---------------------------------

POLICY_OPTIONS	
---------------------------------	---

```
DATABASE_LABELS
----------------------------------------------------------------------
ABCDEFGHIJKLMNOPQRSTUVWXYZ1234 FDF
LBAC$TEST1                                        1 ENABLED


COMPLEX                        RGHU
LBAC$TEST1                                        1 ENABLED


INDIA                          FDFEFG

POLICY_NAME                    COLUMN_NAME
------------------------------ ------------------------------
PACKAGE                          BIN_SIZE STATUS
------------------------------ ----------- --------
DEFAULT_FORMAT                 POLICY_FORMAT
------------------------------ ------------------------------
POLICY_OPTIONS
----------------------------------------------------------------------
DATABASE_LABELS
----------------------------------------------------------------------
LBAC$TEST1                                       1 ENABLED



SILE                           TESTLABEL
LBAC$TEST1                                       1 ENABLED



SIMPLE                         RAGHU
LBAC$TEST1                                       1 ENABLED

POLICY_NAME                    COLUMN_NAME
------------------------------ ------------------------------
PACKAGE                          BIN_SIZE STATUS
------------------------------ ----------- --------
DEFAULT_FORMAT                 POLICY_FORMAT
------------------------------ ------------------------------
POLICY_OPTIONS
----------------------------------------------------------------------
DATABASE_LABELS
----------------------------------------------------------------------

5 rows selected.

[END: CLAIMS 1 and 21]

SQL>
SQL>    EXECUTE LBAC_SYSDBA.ENABLE_POLICY('simple');
[STEP 101: "Simple" policy is enabled; STEPS 101-104 illustrate how it is
determined "whether to perform [an] operation on a row of a table based on a set
of labels associated with the row, the set of labels corresponding to the policy
set", as recited by Claims 1 and 21.]
```

```
PL/SQL procedure successfully completed.

SQL>
SQL> -- The basic objective from now on is to test the enable/disable procedures
...
SQL>
SQL>    EXECUTE LBAC_LABEL_ADMIN.CREATE_LABEL('simple',1,'A,B', TRUE);
[STEP 102: Label "A,B" is associated with policy "simple".]

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_LABEL_ADMIN.CREATE_LABEL('abcdefghijklmnopqrstuvwxyz',2,'A',
TRUE);
BEGIN LBAC_LABEL_ADMIN.CREATE_LABEL('abcdefghijklmnopqrstuvwxyz',2,'A', TRUE);
END;

*
ERROR at line 1:
ORA-12416: policy abcdefghijklmnopqrstuvwxyz not found
ORA-06512: at "LBACSYS.LBAC_CACHE", line .*
ORA-06512: at "LBACSYS.LBAC_LABEL_ADMIN", line .*
ORA-06512: at line 1

SQL>
SQL>    EXECUTE
LBAC_USER_ADMIN.SET_USER_LABELS('abcdefghijklmnopqrstuvwxyz','SCOTT',TO_LABEL_LI
ST.FROM_CHAR('abcdefghijklmnopqrstuvwxyz',NULL,'A'));
BEGIN
LBAC_USER_ADMIN.SET_USER_LABELS('abcdefghijklmnopqrstuvwxyz','SCOTT',TO_LABEL_LI
ST.FROM_CHAR('abcdefghijklmnopqrstuvwxyz',NULL,'A')); END;

*
ERROR at line 1:
ORA-01405: fetched column value is NULL

SQL>
SQL>    SELECT *
  2     FROM DBA_LBAC_USER_LABELS ORDER BY USER_NAME, POLICY_NAME;

no rows selected

SQL>
SQL> -- Error Conditions
SQL>    EXECUTE
LBAC_POLICY_ADMIN.APPLY_TABLE_POLICY('abcdefghijklmnopqrstuvwxyz','SCOTT','abc')
;
BEGIN
LBAC_POLICY_ADMIN.APPLY_TABLE_POLICY('abcdefghijklmnopqrstuvwxyz','SCOTT','abc')
; END;

*
ERROR at line 1:
ORA-12416: policy abcdefghijklmnopqrstuvwxyz not found
ORA-06512: at "LBACSYS.LBAC_CACHE", line .*
ORA-06512: at "LBACSYS.LBAC_POLICY_ADMIN", line .*
ORA-06512: at line 1
```

[START: CLAIMS 1, 2, 5, 21, 22, and 25
For CLAIMS 1 and 21, policy "complex" is applied to jing and policies "complex"
and "simple" are applied to table "abc".
For CLAIMS 2 and 22, a policy column is added when a policy is applied to a
table (e.g. "abc"). (See next bolded section on this page).
For CLAIMS 5 and 25, policy "complex" and "simple" are the two or more policies
of the plurality of label-based policies of table "abc"]
SQL>
SQL> -- OK now
SQL>   EXECUTE
LBAC_POLICY_ADMIN.APPLY_TABLE_POLICY('complex','SCOTT','abc','NO_CONTROL');

PL/SQL procedure successfully completed.

SQL>   EXECUTE
LBAC_POLICY_ADMIN.APPLY_TABLE_POLICY('complex','SCOTT','jing','NO_CONTROL');

PL/SQL procedure successfully completed.

SQL>   EXECUTE
LBAC_POLICY_ADMIN.APPLY_TABLE_POLICY('simple','SCOTT','abc','DELETE_CONTROL');

PL/SQL procedure successfully completed.

[END: CLAIMS 1, 2, 5, 21, 22, and 25]

SQL>   EXECUTE
LBAC_POLICY_ADMIN.APPLY_TABLE_POLICY('simple','SCOTT','EMP','DELETE_CONTROL');

PL/SQL procedure successfully completed.

SQL>   EXECUTE
LBAC_POLICY_ADMIN.APPLY_TABLE_POLICY('sile','SCOTT','jing','DELETE_CONTROL');

PL/SQL procedure successfully completed.

SQL>
SQL> CONNECT SCOTT/TIGER
Connected.
[Step 103: User 'SCOTT' connects to the database. Note: user 'SCOTT is not
associated with any labels.]

[START: CLAIM 2, 5, 22 and 25
For CLAIMS 2 and 22, policy columns "RGHU" and "RAGHU" are added for policies
"complex" and "simple".
For CLAIMS 5 and 25, these columns are added because policies "complex" and
"simple" are applied – i.e. "the policy set associated with the table includes
two or more policies of the plurality of label-based policies."]
SQL>
SQL>   DESC abc;

Name	Null?	Type
COL1		VARCHAR2(45)
RGHU		LBACSYS.LBAC_LABEL
RAGHU		LBACSYS.LBAC_LABEL

[END: CLAIM 2, 5, 22 and 25]

```
SQL>
SQL>    INSERT INTO abc(col1)
  2       VALUES('fdfd');

1 row created.

[START: CLAIMS 3 and 23
Label "A,B" for policy "simple" is stored in policy column "raghu" corresponding
to the policy.]
SQL>
SQL>    UPDATE abc
  2       SET raghu = LBACSYS.TO_LBAC_LABEL('simple','A,B');

1 row updated.

[END: CLAIMS 3 and 23]

[Step 104/START: CLAIMS 1, 4, 21 and 24
For CLAIMS 1 and 21, the following delete operation is received. It was
previously shown which policies apply to table "abc". Here, only policy "simple"
is enforced on delete because DELETE_CONTROL is only specified for policy
"simple" even though both policies "simple" and "complex" are applied. Thus,
policy "simple", of the plurality of label-based policies ("simple" and
"complex"), was determined to be applied to table "abc". In this example, it is
determined that the delete operation is NOT performed based on the set of labels
associated with the row (i.e. user "SCOTT" is not associated with any labels so
the user is denied the ability to delete the row).
For CLAIMS 4 and 24, in order to determine which policies apply, it must be
determined whether a column is a policy column]
SQL> -- Should not allow ...
SQL>    DELETE FROM abc;
   DELETE FROM abc
                 *
ERROR at line 1:
ORA-12406: unauthorized SQL statement for policy SIMPLE
ORA-06512: at "LBACSYS.LBAC_STANDARD", line 0
ORA-06512: at "LBACSYS.LBAC.*
ORA-04088: error during execution of trigger 'LBACSYS.LBAC.*
[END: CLAIMS 1, 4, 21, and 24]

[START: CLAIMS 3 and 23
Shows that label "A,B" associated with policy "simple" is stored in policy
column "raghu" of a row, in table "abc", with value "fdfd"]
SQL>
SQL>    SELECT col1,LABEL_TO_CHAR(raghu)
  2       FROM abc
  3       ORDER BY col1;

COL1
------------------------------------------------------------
LABEL_TO_CHAR(RAGHU)
--------------------------------------------------------------------------------
fdfd
A,B

1 row selected.
[END: CLAIMS 3 and 23]
```

```
SQL>
SQL> -- Error Condition
SQL>    EXECUTE LBAC_SYSDBA.DISABLE_POLICY('simple');
BEGIN LBAC_SYSDBA.DISABLE_POLICY('simple'); END;

      *
ERROR at line 1:
ORA-06550: line 1, column 7:
PLS-00201: identifier 'LBACSYS.LBAC_SYSDBA' must be declared
ORA-06550: line 1, column 7:
PL/SQL: Statement ignored


SQL>
SQL> CONNECT LBACSYS/LBACSYS
Connected.
SQL>
SQL>    EXECUTE LBAC_SYSDBA.DISABLE_POLICY('simple');

PL/SQL procedure successfully completed.

SQL>
SQL> -- Error Conditions ...
SQL>    EXECUTE LBAC_SYSDBA.DISABLE_POLICY('abcdefghijklmnopqrstuvwxyzfd');
BEGIN LBAC_SYSDBA.DISABLE_POLICY('abcdefghijklmnopqrstuvwxyzfd'); END;

*
ERROR at line 1:
ORA-12416: policy abcdefghijklmnopqrstuvwxyzfd not found
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-06512: at line 1


SQL>    EXECUTE LBAC_SYSDBA.DISABLE_POLICY('abcdefghijklmnopqrstuvwxyz','fdf');
BEGIN LBAC_SYSDBA.DISABLE_POLICY('abcdefghijklmnopqrstuvwxyz','fdf'); END;

      *
ERROR at line 1:
ORA-06550: line 1, column 7:
PLS-00306: wrong number or types of arguments in call to 'DISABLE_POLICY'
ORA-06550: line 1, column 7:
PL/SQL: Statement ignored

SQL>
SQL> -- Should not delete as the disable will be effective from next session
only
SQL>    DELETE FROM SCOTT.abc;

1 row deleted.

SQL>
SQL> CONNECT SCOTT/TIGER
Connected.
SQL>
SQL>    SELECT col1,LABEL_TO_CHAR(raghu)
   2    FROM abc
   3    ORDER BY col1;
```

```
no rows selected

SQL>
SQL> -- Should delete now as the policy is disabled ...
SQL>    DELETE FROM abc;

0 rows deleted.

SQL>
SQL>    SELECT col1,LABEL_TO_CHAR(raghu)
  2      FROM abc
  3      ORDER BY col1;

no rows selected

SQL>
SQL>    INSERT INTO abc(col1)
  2      VALUES('123233');

1 row created.

SQL>
SQL> -- Error Condition ...
SQL>    EXECUTE LBAC_SYSDBA.ENABLE_POLICY('simple');
BEGIN LBAC_SYSDBA.ENABLE_POLICY('simple'); END;

        *
ERROR at line 1:
ORA-06550: line 1, column 7:
PLS-00201: identifier 'LBACSYS.LBAC_SYSDBA' must be declared
ORA-06550: line 1, column 7:
PL/SQL: Statement ignored


SQL>
SQL> CONNECT LBACSYS/LBACSYS
Connected.
SQL>
SQL>    EXECUTE LBAC_SYSDBA.ENABLE_POLICY('simple');

PL/SQL procedure successfully completed.

SQL>
SQL> -- Error Conditions ...
SQL>    EXECUTE LBAC_SYSDBA.ENABLE_POLICY('simpler1');
BEGIN LBAC_SYSDBA.ENABLE_POLICY('simpler1'); END;

*
ERROR at line 1:
ORA-12416: policy simpler1 not found
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-06512: at line 1


SQL>    EXECUTE LBAC_SYSDBA.ENABLE_POLICY('simple',FALSE);
BEGIN LBAC_SYSDBA.ENABLE_POLICY('simple',FALSE); END;
```

```
                    *
ERROR at line 1:
ORA-06550: line 1, column 7:
PLS-00306: wrong number or types of arguments in call to 'ENABLE_POLICY'
ORA-06550: line 1, column 7:
PL/SQL: Statement ignored

SQL>
SQL> -- Should delete now as the enable will be effective only from new session
SQL>    DELETE FROM SCOTT.abc;

1 row deleted.

SQL>
SQL> CONNECT SCOTT/TIGER
Connected.
SQL>
SQL> -- Expecting no rows ...
SQL>    SELECT col1,LABEL_TO_CHAR(raghu)
  2       FROM abc
  3       ORDER BY col1;

no rows selected

SQL>
SQL>    INSERT INTO abc(col1)
  2       VALUES('1232');

1 row created.

SQL>
SQL> -- Delete should fail ...
SQL>    DELETE FROM abc;
   DELETE FROM abc
                   *
ERROR at line 1:
ORA-12406: unauthorized SQL statement for policy SIMPLE
ORA-06512: at "LBACSYS.LBAC_STANDARD", line 0
ORA-06512: at "LBACSYS.LBAC.*
ORA-04088: error during execution of trigger 'LBACSYS.LBAC.*


SQL>
SQL>    SELECT col1,LABEL_TO_CHAR(raghu)
  2       FROM abc
  3       ORDER BY col1;

COL1
------------------------------------------------
LABEL_TO_CHAR(RAGHU)
-------------------------------------------------------------------------------
1232


1 row selected.
```

```
SQL>
SQL> CONNECT LBACSYS/LBACSYS
Connected.
SQL>
SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('simple',TRUE);

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('complex', FALSE);

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('sile');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('abcdefghijklmnopqrstuvwxyz1234');

PL/SQL procedure successfully completed.

SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('abcdefghijklmnopqrstuvwxyz');
BEGIN LBAC_SYSDBA.DROP_POLICY('abcdefghijklmnopqrstuvwxyz'); END;

*
ERROR at line 1:
ORA-12416: policy abcdefghijklmnopqrstuvwxyz not found
ORA-06512: at "LBACSYS.LBAC_STANDARD", line 0
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-06512: at line 1

SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('india');

PL/SQL procedure successfully completed.

SQL>
SQL> -- Error Conditions
SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('adfd');
BEGIN LBAC_SYSDBA.DROP_POLICY('adfd'); END;

*
ERROR at line 1:
ORA-12416: policy adfd not found
ORA-06512: at "LBACSYS.LBAC_STANDARD", line 0
ORA-06512: at "LBACSYS.LBAC_SYSDBA", line .*
ORA-06512: at line 1

SQL>    EXECUTE LBAC_SYSDBA.DROP_POLICY('simple',XYZ);
BEGIN LBAC_SYSDBA.DROP_POLICY('simple',XYZ); END;
                                         *
ERROR at line 1:
ORA-06550: line 1, column 40:
PLS-00201: identifier 'XYZ' must be declared
ORA-06550: line 1, column 7:
PL/SQL: Statement ignored


SQL>
```

```
SQL>    SELECT *
  2     FROM DBA_LBAC_POLICIES
  3     ORDER BY POLICY_NAME;

no rows selected

SQL>
SQL> CONNECT SCOTT/TIGER
Connected.
SQL>
SQL> -- Simple policy was applied on two tables (abc,emp) the hidden column
should
SQL> -- be dropped as the TRUE option is set; Policy complex was applied to
SQL> -- two tables(abc,jing) and hidden column should not be dropped as the
option
SQL> -- was set to FALSE; Policy sile was applied to a table (jing) and the
hidden
SQL> -- column should not be dropped as the default option is FALSE.
SQL>
SQL>    DESC abc;
 Name                                          Null?     Type
 --------------------------------------------- --------- ----------------------------
 COL1                                                    VARCHAR2(45)
 RGHU                                                    LBACSYS.LBAC_LABEL

SQL>    DESC jing;
 Name                                          Null?     Type
 --------------------------------------------- --------- ----------------------------
 COL1                                                    VARCHAR2(45)
 RGHU                                                    LBACSYS.LBAC_LABEL
 TESTLABEL                                               LBACSYS.LBAC_LABEL

SQL>    DESC EMP;
 Name                                          Null?     Type
 --------------------------------------------- --------- ----------------------------
 EMPNO                                         NOT NULL  NUMBER(4)
 ENAME                                                   VARCHAR2(10)
 JOB                                                     VARCHAR2(9)
 MGR                                                     NUMBER(4)
 HIREDATE                                                DATE
 SAL                                                     NUMBER(7,2)
 COMM                                                    NUMBER(7,2)
 DEPTNO                                                  NUMBER(2)

SQL>
SQL>    DROP TABLE abc;

Table dropped.

SQL>    DROP TABLE jing;

Table dropped.

SQL>
SQL> SET ECHO OFF
```